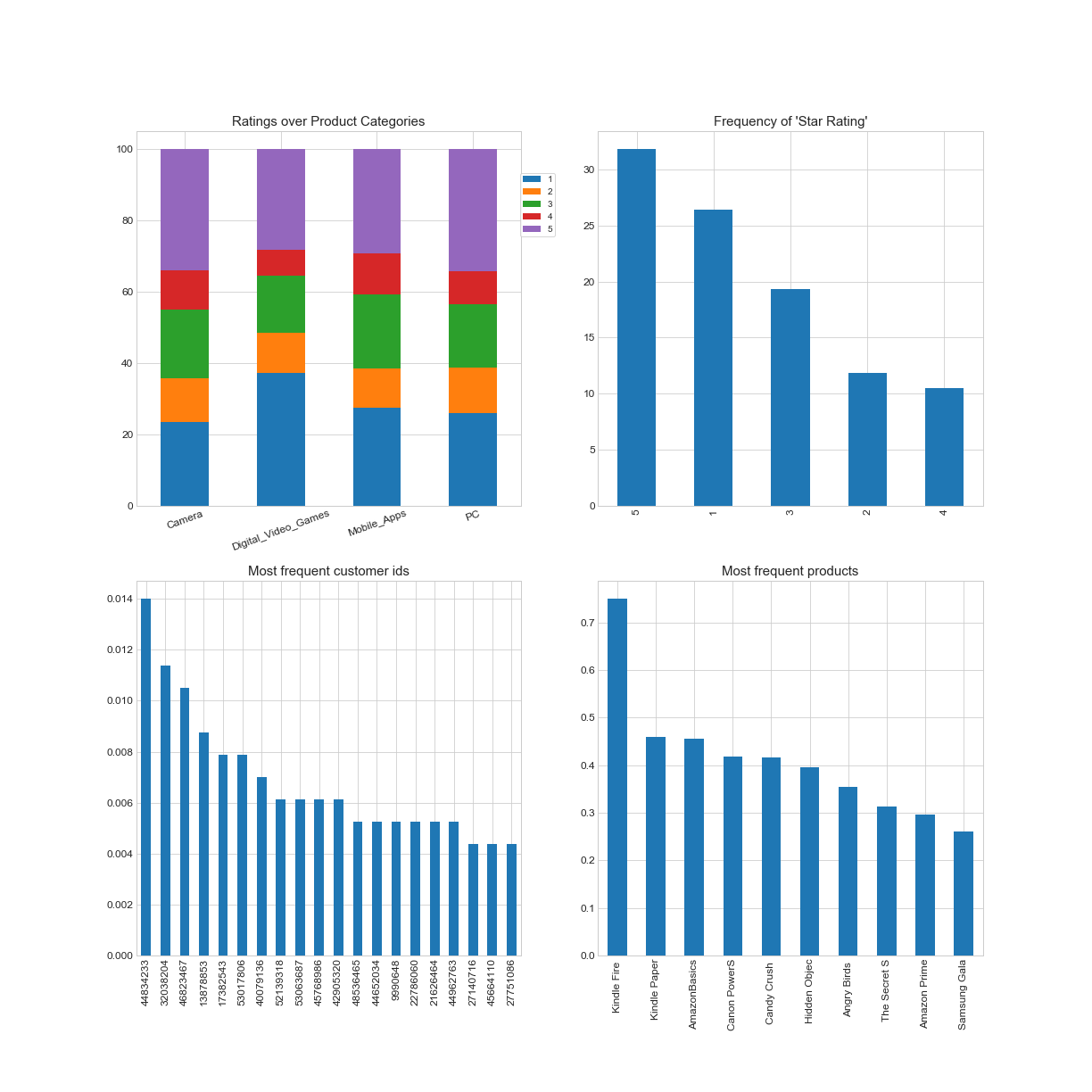
Milestone Report

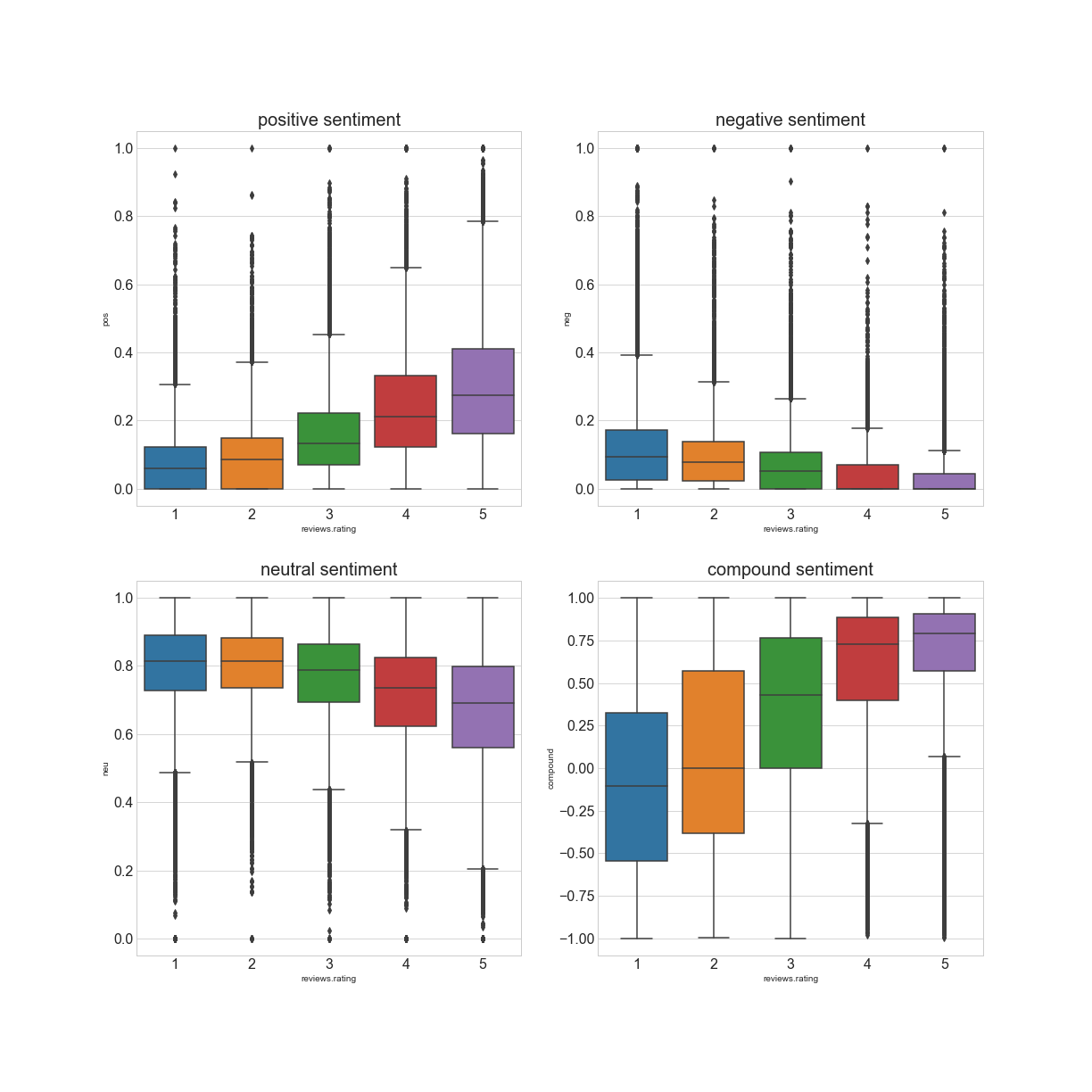
* **Business Problem & Approach:**

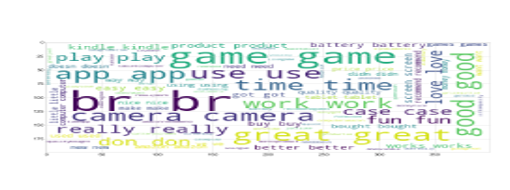
Many enterprises which are challenged to improve their products and services, either if they are sold online or in a more traditional way, need to rely on interaction with and feedback from their customers. Hence, the customer’s “voice” about the service or product might be of extensive value in order to re-design the way a company is marketing and servicing its products or it might provide important information for the development for new product series. Therefore, extracting and summarizing information of product reviews is key for many companies to remain competitive. However, many NLP applications focus on classifying reviews to be either positive or negative or do provide another kind of sentiment quantification. In turn, these measures are important to summarize the impression and feelings of customers over a wide range of products. Nevertheless, classifying reviews with regard to sentiment patterns might not provide the sufficient information for further improvements of products, because useful information about details or key problems customers are facing is getting lost. Thus, besides classifying reviews, this project aims to develop a method to summarize product reviews. For instance, it would be quite helpful for a product or marketing manager to get a summary of the 5000 worst feedbacks of a product. Moreover, from another angle, it could be also quite useful to get a summary of the 5000 best feedbacks in order to figure out “unique selling propositions (USP)”. The approach to solve above targets is twofold. First, Deep Learning is applied to the solve the classification task, i.e. to predict the ratings a review gives a product. In a second step, extractive summarization techniques get developed leveraging the encodings provided by the network. Thus, the model of this project should be able to 1) classify a review and 2) to provide a framework to generate summaries of certain reviews which could get selected by a user. Basically, step 1 is necessary because many companies might not have a clear functionality for customers to rate products on a scale, moreover, in other applications – where the company relies, for example, on datasets like email messaging form clients, there are not any possibilities to give categorical ratings without the need of a human. Thus, step 1 provides an important output which then helps to segment reviews without the need of labelling the data beforehand in a company. Step 2 is basically leveraging the Deep Learning Classifier’s embeddings to generate summaries by means of most representative sentences or reviews.

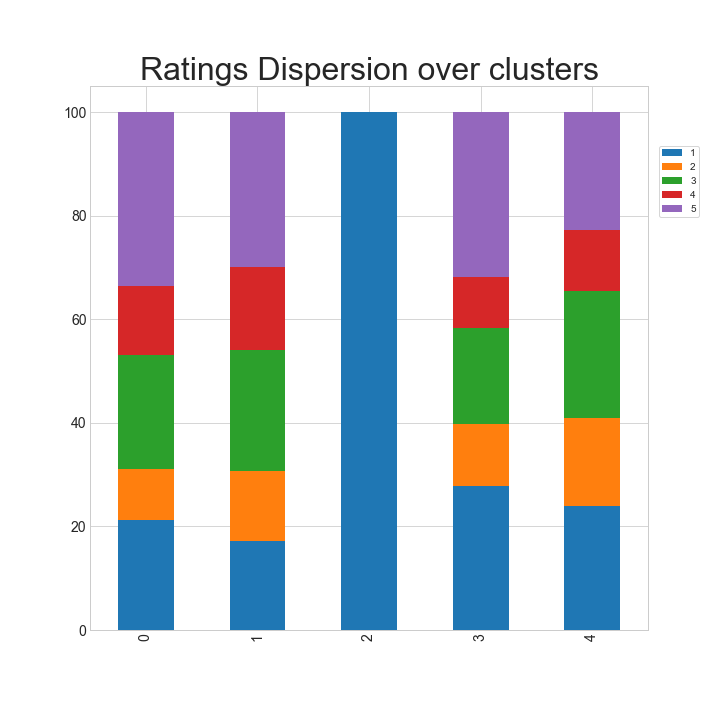
* **Data & EDA**

The dataset has been created by the author to generate a representative subset of Amazon reviews. Following the page <https://s3.amazonaws.com/amazon-reviews-pds/tsv/index.txt> which provides links to reviews of diverse categories hosted by Amazon, reviews about “Mobile Apps”, “PC”, “Camera” and “Digital Video Games have been downloaded. These categories could be subsumed to an overall product topic like “Digital Entertainment and Electronic Products”. After dropping duplicates and a small number of three NA values the created dataset contains 114192 reviews with the proportions shown in the below chart. 

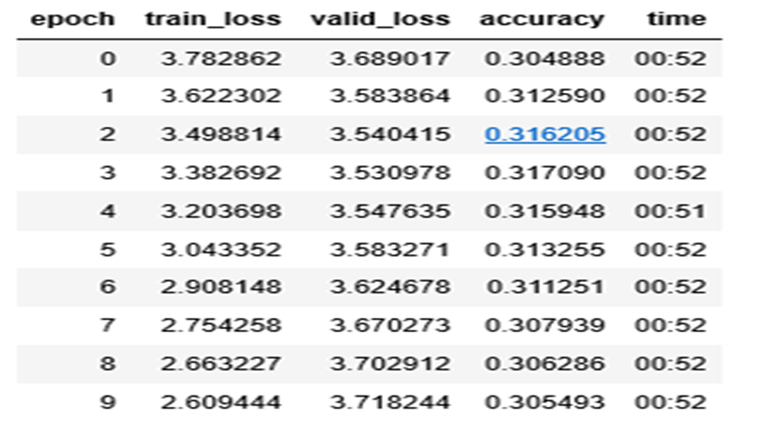
* : 5 and 4 stars make up more than 40 percent of all reviews and the lower end – 2 and 1 star ratings - sum up to close to 40 percent as well. The neutral ratings – 3 star ratings – show a comparably small frequency
* “Kindle Fire” has been most often reviewed, followed by “Kindle Paper” and “Amazon Basics”

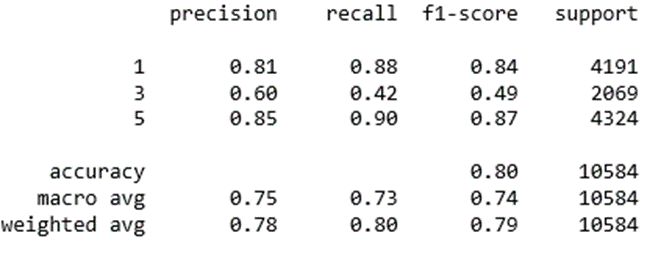


* Lexical approach to classify reviews’ sentiment: , neutral scores are comparably higher for lower rated reviews which might hint at some kind of behavioural bias, i.e. people could be quite dissatisfied with a product but might express their view with more neutral wordings
* **Cluster Analysis and Topic extraction:**
* **Most frequently used words**
* **Clustering through NMF provides 5 groups**



First two clusters more on positive side, third one entirely negative – break point to rather bad rating proportions

* **Deep Learning:**
* ULMFit Language Model: key to use pretrained embeddings and to refine them on 60 K reviews. Target of learner to predict the next word given the previous word. Accuracy is not key, but the generation of context specific word embeddings; as output words of reviews are encoded in a 400 dimensional vector 
* Deep Learning Classifier: changing the last layers to dense ones for classification. Basically, to focus on polarity and to avoid “learning” patterns due to behavioural biases top and bottom ratings are subsumed to tow classes, i.e. (5/4 star -> 5) and (1/2 star -> 1), while it 3 star category remains. Learning process is gradually unfreezing layers while training. Results are satisfactory, though, most misclassifications are those of the middle/neutral category



* **Review Extraction:** aggregating word embeddings from language model by averaging on a review level to achieve 400 dimensional representations of reviews. Clustering is used and the closest reviews are selected to be most representative.